

AUS920030674US1

Patent Application

NAVIGATING A UAV WITH TELEMETRY THROUGH A SOCKET

Inventors: William Kress Bodin

5. Jesse J.W. Redman

Derral C. Thorson

BACKGROUND OF THE INVENTION

10

Field of the Invention

15 The field of the invention is data processing, or, more specifically, methods, systems, and products for navigating a UAV.

Description Of Related Art

20 Many forms of UAV are available in prior art, both domestically and internationally. Their payload weight carrying capability, their accommodations (volume, environment), their mission profiles (altitude, range, duration), and their command, control and data acquisition capabilities vary significantly. Routine civil access to these various UAV assets is in an embryonic state.

25 Conventional UAVs are typically manually controlled by an operator who may view aspects of a UAV's flight using cameras installed on the UAV with images provided through downlink telemetry. Navigating such UAVs from a starting position to one

or more waypoints requires an operator to have specific knowledge of the UAV's flight, including such aspects as starting location, the UAV's current location, waypoint locations, and so on. Operators of prior art UAVs usually are required generally to manually control the UAV from a starting position to a waypoint with  
5 little aid from automation. There is therefore an ongoing need for improvement in the area of UAV navigations.

SUMMARY OF THE INVENTION

Methods, systems, and products are described for UAV navigation that enable an operator to input a single interface operation, a mouseclick or joystick button click, for example, thereby selecting a GUI pixel from a displayed GUI map of the surface of the Earth. The selected pixel maps to a waypoint. The waypoint coordinates are uploaded through uplink telemetry to a UAV which calculates a heading and flies, according to a navigation algorithm, a course to the waypoint. Alternatively, a remote control device can calculate a heading and uplink flight control instructions to a UAV. The heading is not necessarily the course if wind is present, depending on the navigation algorithm chosen for the flight. All this occurs with a single keystroke or mouseclick from the operator.

The operator's remote control device from which the pixel is selected is enabled according to some embodiments of the present invention to be very thin. Often the remote control device can be a browser in a laptop or personal computer or a microbrowser in a PDA enhanced only with client-side scripting sufficient to map a pixel to a waypoint and transmit the waypoint to the UAV. The UAV itself in such embodiments may comprise the intelligence, the navigation algorithms, a web server to download map images to a client browser in a remote control device, a repository of Landsat maps from which HTML screens are formulated for download to the remote control device, and so on. Alternatively, navigation algorithms and Landsat maps or other satellite imaging maps may reside on a more intelligent remote control device that uplinks flight control instructions to a UAV whose on-board computing capacity is relatively thin.

In addition to navigating with a single waypoint, operators of remote control devices

according to embodiments of the present invention are enabled to enter through a user interface and upload to the UAV many waypoints which taken in sequence form an entire mission for a UAV that flies from waypoint to waypoint. In addition to providing for a mission route comprising many waypoints, typical embodiments also support 'macros,' sets of UAV instructions associated with waypoints. Such UAV instructions can include, for example, instructions to orbit, take photographs or stream video, and continue flying a route or mission to a next waypoint. Because waypoints are entered with selected pixels and macros may be created by selecting UAV instructions from a pull down menu in a GUI, complex missions may be established with a few keystrokes of mouseclicks on an interface of a remote control device. In embodiments where the waypoints and UAV instructions are uploaded and stored on the UAV along with the navigation algorithms needed to travel from waypoint to waypoint, the remote control device may lose communications with the UAV or even be destroyed completely, and the UAV will simply continue its mission.

More particularly, methods, systems, and products are disclosed in this specification for navigating a UAV. Typical embodiments include receiving in a remote control device a user's selection of a GUI map pixel that represents a waypoint for UAV navigation, the pixel having a location on the GUI; mapping the pixel's location on the GUI to Earth coordinates of the waypoint; receiving downlink telemetry, including a starting position from a GPS receiver on the UAV, from the UAV through a socket on the remote control device; calculating a heading in dependence upon the starting position, the coordinates of the waypoint, and a navigation algorithm; identifying flight control instructions for flying the UAV on the heading; and transmitting uplink telemetry, including the flight control instructions, through the socket to the UAV.

In typical embodiments, receiving downlink telemetry also includes listening on the socket for downlink data; storing downlink data in computer memory; and exposing the stored downlink data through an API to a navigation application. In typical embodiments, the downlink telemetry further comprises flight control instructions. In  
5 typical embodiments, The transmitting uplink telemetry further comprises monitoring computer memory for uplink data from a navigation application and, when uplink data is presented, sending the uplink data through the socket to the UAV.

Many embodiments also include receiving user selections of a multiplicity of GUI  
10 map pixels representing waypoints, each pixel having a location on the GUI; mapping each pixel location to Earth coordinates of a waypoint; assigning one or more UAV instructions to each waypoint; storing the coordinates of the waypoints and UAV instructions in computer memory on the remote control device; flying the UAV to each waypoint in accordance with one or more navigation algorithms; and operating  
15 the UAV at each waypoint in accordance with the UAV instructions for each waypoint. In such embodiments, operating the UAV at each waypoint in accordance with the UAV instructions also includes identifying flight control instructions in dependence upon the UAV instructions for each waypoint and transmitting the flight control instructions in the uplink telemetry through the socket from the remote control  
20 device to the UAV.

In many embodiments, mapping the pixel's location on the GUI to Earth coordinates of the waypoint also include mapping pixel boundaries of the GUI map to Earth coordinates; identifying a range of latitude and a range of longitude represented by  
25 each pixel; and locating a region on the surface of the Earth in dependence upon the boundaries, the ranges, and the location of the pixel on the GUI map. In many embodiments, locating a region on the surface of the Earth in dependence upon the

boundaries, the ranges, and the location of the pixel on the GUI map is carried out by multiplying the range of longitude represented by each pixel by a column number of the selected pixel, yielding a first multiplicand; multiplying the range of longitude represented by each pixel by 0.5, yielding a second multiplicand; adding the first and  
5 second multiplicands to an origin longitude of the GUI map; multiplying the range of latitude represented by each pixel by a row number of the selected pixel, yielding a third multiplicand; multiplying the range of latitude represented by each pixel by 0.5, yielding a fourth multiplicand; and adding the third and fourth multiplicands to an origin latitude of the GUI map.

10

Also disclosed in this specification are methods, systems, and products for navigating a UAV that include receiving in a remote control device a user's selection of a GUI map pixel that represents a waypoint for UAV navigation, the pixel having a location on the GUI; mapping the pixel's location on the GUI to Earth coordinates of the  
15 waypoint; transmitting uplink telemetry, including the coordinates of the waypoint, to the UAV through a socket on the remote control device; receiving downlink telemetry, include a starting position from a GPS receiver, from the UAV through the socket; and piloting the UAV, under control of a navigation computer on the UAV, from the starting position to the waypoint in accordance with a navigation algorithm.

20

In such embodiments, transmitting uplink telemetry typically also includes monitoring computer memory for uplink data from a navigation application and, when uplink data is presented, sending the uplink data through the socket to the UAV. In such embodiments, receiving downlink telemetry is typically carried out by  
25 listening on the socket for downlink data; storing downlink data in computer memory; and exposing the stored downlink data through an API to a navigation application. In many embodiments, downlink telemetry may include Earth coordinates of waypoints

or UAV instructions.

Typical embodiments also include receiving user selections of a multiplicity of GUI map pixels representing waypoints, each pixel having a location on the GUI; mapping  
5 each pixel location to Earth coordinates of a waypoint; assigning one or more UAV instructions to each waypoint; transmitting the coordinates of the waypoints and the UAV instructions in the uplink telemetry through the socket to the UAV; storing the coordinates of the waypoints and UAV instructions in computer memory on the UAV; piloting the UAV to each waypoint in accordance with one or more navigation  
10 algorithms; and operating the UAV at each waypoint in accordance with the UAV instructions for each waypoint.

In such embodiments, mapping the pixel's location on the GUI to Earth coordinates of the waypoint is carried out by mapping pixel boundaries of the GUI map to Earth  
15 coordinates; identifying a range of latitude and a range of longitude represented by each pixel; and locating a region on the surface of the Earth in dependence upon the boundaries, the ranges, and the location of the pixel on the GUI map. In such embodiments, locating a region on the surface of the Earth in dependence upon the boundaries, the ranges, and the location of the pixel on the GUI map typically is  
20 carried out by multiplying the range of longitude represented by each pixel by a column number of the selected pixel, yielding a first multiplicand; multiplying the range of longitude represented by each pixel by 0.5, yielding a second multiplicand; adding the first and second multiplicands to an origin longitude of the GUI map; multiplying the range of latitude represented by each pixel by a row number of the  
25 selected pixel, yielding a third multiplicand; multiplying the range of latitude represented by each pixel by 0.5, yielding a fourth multiplicand; and adding the third and fourth multiplicands to an origin latitude of the GUI map.

The foregoing and other objects, features and advantages of the invention will be apparent from the following more particular descriptions of exemplary embodiments of the invention as illustrated in the accompanying drawings wherein like reference  
5 numbers generally represent like parts of exemplary embodiments of the invention.



BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 sets forth a system diagram illustrating relations among components of an exemplary system for navigating a UAV.

5

Figure 2 is a block diagram of an exemplary UAV showing relations among components of included automated computing machinery.

Figure 3 is a block diagram of an exemplary remote control device showing relations among components of included automated computing machinery.

10

Figure 4 sets forth a flow chart illustrating an exemplary method for navigating a UAV that includes receiving in a remote control device a user's selection of a GUI map pixel that represents a waypoint for UAV navigation.

15

Figure 4A is a data flow diagram illustrating an exemplary method for receiving downlink telemetry.

Figure 4B sets forth a data flow diagram illustrating an exemplary method for transmitting uplink telemetry.

20

Figure 5 sets forth a block diagram that includes a GUI displaying a map and a corresponding area of the surface of the Earth.

Figure 6 sets forth a flow chart illustrating an exemplary method of navigating a UAV in accordance with a navigation algorithm.

25

Figure 7 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 6.

5 Figure 8 sets forth a flow chart illustrating an exemplary method of navigating a UAV in accordance with a navigation algorithm.

Figure 9 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 8.

10 Figure 10 sets forth a flow chart illustrating an exemplary method of navigating a UAV in accordance with a navigation algorithm.

Figure 11 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 10.

15

Figure 12 sets forth a flow chart illustrating an exemplary method for navigating a UAV that includes receiving in a remote control device a user's selection of a GUI map pixel that represents a waypoint for UAV navigation.

20 Figure 13 sets forth a flow chart illustrating an exemplary method of piloting in accordance with a navigation algorithm.

Figure 14 sets forth a line drawing illustrating a method of calculating a heading with a cross wind to achieve a particular ground course.

25

Figure 15 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 13.

Figure 16 sets forth a flow chart illustrating an exemplary method of piloting in accordance with a navigation algorithm.

- 5    Figure 17 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 16.

DETAILED DESCRIPTION OF EXEMPLARY EMBODIMENTSIntroduction

5 The present invention is described to a large extent in this specification in terms of methods for navigating a UAV. Persons skilled in the art, however, will recognize that any computer system that includes suitable programming means for operating in accordance with the disclosed methods also falls well within the scope of the present invention. Suitable programming means include any means for directing a computer  
10 system to execute the steps of the method of the invention, including for example, systems included of processing units and arithmetic-logic circuits coupled to computer memory, which systems have the capability of storing in computer memory, which computer memory includes electronic circuits configured to store data and program instructions, programmed steps of the method of the invention for  
15 execution by a processing unit.

The invention also may be embodied in a computer program product, such as a diskette or other recording medium, for use with any suitable data processing system. Embodiments of a computer program product may be implemented by use of any  
20 recording medium for machine-readable information, including magnetic media, optical media, or other suitable media. Persons skilled in the art will immediately recognize that any computer system having suitable programming means will be capable of executing the steps of the method of the invention as embodied in a program product. Persons skilled in the art will recognize immediately that, although  
25 most of the exemplary embodiments described in this specification are oriented to software installed and executing on computer hardware, nevertheless, alternative embodiments implemented as firmware or as hardware are well within the scope of

the present invention.

### Definitions

5 “Air speed” means UAV air speed, the speed of the UAV through the air.

A “cross track” is a fixed course from a starting point directly to a waypoint. A cross track has a direction, a ‘cross track direction,’ that is the direction straight from a starting point to a waypoint. That is, a cross track direction is the heading that a UAV  
10 would fly directly from a starting point to a waypoint in the absence of wind.

“GUI” means graphical user interface, a display means for a computer screen.

“Heading” means the compass heading of the UAV. “Course” means the direction of  
15 travel of the UAV over the ground. In the absence of wind, or in the presence of a straight tailwind or straight headwind, the course and the heading are the same direction. In the presence of crosswind, the course and the heading are different directions.

20 “Position” refers to a location in the air or over the ground. ‘Position’ is typically specified as Earth coordinates, latitude and longitude. A specification of position may also include altitude.

A “waypoint” is a position chosen as a destination for navigation of a route. A route  
25 has one or more waypoints. That is, a route is composed of waypoints, including at least one final waypoint, and one or more intermediate waypoints.

“TDMA” stands for Time Division Multiple Access, a technology for delivering digital wireless service using time-division multiplexing. TDMA works by dividing a radio frequency into time slots and then allocating slots to multiple calls. In this way, a single frequency can support multiple, simultaneous data channels. TDMA is used  
5 by GSM.

“GSM” stands for Global System for Mobile Communications, a digital cellular standard. GSM at this time is the de facto standard for wireless digital communications in Europe and Asia.

10

“CDPD” stands for Cellular Digital Packet Data, a data transmission technology developed for use on cellular phone frequencies. CDPD uses unused cellular channels to transmit data in packets. CDPD supports data transfer rates of up to 19.2 Kbps.

15

“GPRS” stands for General Packet Radio Service, a standard for wireless data communications which runs at speeds up to 150 Kbps, compared with current GSM systems which cannot support more than about 9.6 Kbps. GPRS, which supports a wide range of speeds, is an efficient use of limited bandwidth and is particularly  
20 suited for sending and receiving small bursts of data, such as e-mail and Web browsing, as well as large volumes of data.

“EDGE” stands for Enhanced Data Rates for GSM Evolution, a standard for wireless data communications supporting data transfer rates of more than 300 Kbps. GPRS  
25 and EDGE are considered interim steps on the road to UMTS.

“UMTS” stands for Universal Mobile Telecommunication System, a standard for wireless data communications supporting data transfer rates of up to 2 Mbps. UMTS is also referred to W-CDMA for Wideband Code Division Multiple Access.

5                                    Navigating a UAV with a Telemetry Through a Socket

Methods, systems, and products for navigating a UAV are explained with reference to the accompanying drawings, beginning with Figure 1. Figure 1 sets forth a system diagram illustrating relations among components of an exemplary system for  
10    navigating a UAV. The system of Figure 1 includes UAV (100) which includes a GPS (Global Positioning System) receiver (not shown) that receives a steady stream of GPS data from satellites (190, 192). For convenience of explanation, only two GPS satellites are shown in Figure 1, although the GPS satellite network in fact includes 24 GPS satellites.

15    The system of Figure 1 operates to navigate a UAV by receiving in a remote control device a user’s selection of a GUI map pixel that represents a waypoint for UAV navigation. Each such pixel has a location on a GUI map, typically specified as a row and column position. Examples of remote control devices in Figure 1 include mobile  
20    telephone (110), workstation (104), laptop computer (116), and PDA (Personal Digital Assistant) (120). Each such remote control device is capable of supporting a GUI display of a map of the surface of the Earth in which each pixel on the GUI map represents a position on the Earth.

25    Each remote control device also supports at least one user input device through which a user may enter the user’s selection of a pixel. Examples of user input devices in the system of Figure 1 include telephone keypad (122), workstation keyboard (114),

workstation joystick (112), laptop keyboard (116) and PDA touch screen (118).

The system of Figure 1 typically is capable of operating a remote control device to map the pixel's location on the GUI to Earth coordinates of a waypoint. The remote control device is often capable of receiving downlink telemetry including starting position from a GPS receiver on the UAV through the socket. In fact, the remote control device is often receiving downlink telemetry that includes a steady stream of GPS positions of the UAV. Receiving a starting position therefore is typically carried out by taking the current position of the UAV when the user selects the pixel as the starting position. In the example of Figure 1, the remote control device generally receives the starting position from the UAV through wireless network (102). The remote control device is often capable of transmitting uplink telemetry including the coordinates of the waypoint, flight control instructions, or UAV instructions through a socket on the remote control devices.

Wireless network (102) is implemented using any wireless data transmission technology as will occur to those of skill in the art including, for example, TDMA, GSM, CDPD, GPRS, EDGE, and UMTS. In a one embodiment, a data communications link layer is implemented using one of these technologies, a data communications network layer is implemented with the Internet Protocol ("IP"), and a data communications transmission layer is implemented using the Transmission Control Protocol ("TCP"). In such systems, telemetry between the UAV and remote control devices, including starting positions, UAV instructions, and flight control instructions, are transmitted using an application-level protocol such as, for example, the HyperText Transmission Protocol ("HTTP"), the Wireless Application Protocol ("WAP"), the Handheld Device Transmission Protocol ("HDTP"), or any other data communications protocol as will occur to those of skill in the art.



The system of Figure 1 typically is capable of calculating a heading in dependence upon the starting position, the coordinates of the waypoint, and a navigation algorithm, identifying flight control instructions for flying the UAV on the heading, and transmitting (420) the flight control instructions from the remote control device to the UAV.

UAVs according to embodiments of the present invention typically include, not only an aircraft, but also automated computing machinery capable of receiving GPS data, operating telemetry between the UAV and one or more remote control devices, and navigating a UAV among waypoints. Figure 2 is a block diagram of an exemplary UAV showing relations among components of included automated computing machinery. In Figure 2, UAV (100) includes a processor (164), also typically referred to as a central processing unit or 'CPU.' The processor may be a microprocessor, a programmable control unit, or any other form of processor useful according to the form factor of a particular UAV as will occur to those of skill in the art. Other components of UAV (100) are coupled for data transfer to processor (164) through system bus (100).

UAV (100) includes random access memory or 'RAM' (166). Stored in RAM (166) is an application program (158) that implements inventive methods according to embodiments of the present invention. In some embodiments, the application programming runs on an OSGi services framework (156). OSGi Stands for 'Open Services Gateway Initiative.' The OSGi specification is a Java-based application layer framework that provides vendor neutral application layer APIs and functions. An OSGi service framework (126) is written in Java and therefore typically runs on a Java Virtual Machine (JVM) (154) which in turn runs on an operating system (150).

Examples of operating systems useful in UAVs according to the present invention include Unix, AIX<sub>TM</sub>, and Microsoft Windows<sub>TM</sub>.

In OSGi, the framework is a hosting platform for running 'services'. Services are the  
5 main building blocks for creating applications according to the OSGi. A service is a group of Java classes and interfaces that implement a certain feature. The OSGi specification provides a number of standard services. For example, OSGi provides a standard HTTP service that can respond to requests from HTTP clients, such as, for example, remote control devices according to embodiments of the present invention.  
10 That is, such remote control devices are enabled to communicate with a UAV having an HTTP service by use of data communications messages in the HTTP protocol.

Services in OSGi are packaged in 'bundles' with other files, images, and resources that the services need for execution. A bundle is a Java archive or 'JAR' file  
15 including one or more service implementations, an activator class, and a manifest file. An activator class is a Java class that the service framework uses to start and stop a bundle. A manifest file is a standard text file that describes the contents of the bundle.

20 The services framework in OSGi also includes a service registry. The service registry includes a service registration including the service's name and an instance of a class that implements the service for each bundle installed on the framework and registered with the service registry. A bundle may request services that are not included in the bundle, but are registered on the framework service registry. To find a service, a  
25 bundle performs a query on the framework's service registry.

In the UAV (100) of Figure 2, software programs and other useful information may

be stored in RAM or in non-volatile memory (168). Non-volatile memory (168) may be implemented as a magnetic disk drive such as a micro-drive, an optical disk drive, static read only memory ('ROM'), electrically erasable programmable read-only memory space ('EEPROM' or 'flash' memory), or otherwise as will occur to those of skill in the art.

UAV (100) includes communications adapter (170) implementing data communications connections (184) to other computers (162), which may be wireless networks, satellites, remote control devices, servers, or others as will occur to those of skill in the art. Communications adapter (170) advantageously facilitates receiving flight control instructions from a remote control device. Communications adapters implement the hardware level of data communications connections through which UAVs transmit wireless data communications. Examples of communications adapters include wireless modems for dial-up connections through wireless telephone networks.

UAV (100) includes servos (178). Servos (178) are proportional control servos that convert digital control signals from system bus (160) into actual proportional displacement of flight control surfaces, ailerons, elevators, and the rudder. The displacement of flight control surfaces is 'proportional' to values of digital control signals, as opposed to the 'all or nothing' motion produced by some servos. In this way, ailerons, for example, may be set to thirty degrees, sixty degrees, or any other supported angle rather than always being only neutral or fully rotated. Several proportional control servos useful in various UAVs according to embodiments of the present invention are available from Futaba®.

UAV (100) includes a servo control adapter (172). A servo control adapter (172) is

multi-function input/output servo motion controller capable of controlling several servos. An example of such a servo control adapter is the "IOSERVO" model from National Control Devices of Osceola, Missouri. The IOSERVO is described on National Control Devices website at [www.controlanything.com](http://www.controlanything.com).

5

UAV (100) includes a flight stabilizer system (174). A flight stabilizer system is a control module that operates servos (178) to automatically return a UAV to straight and level flight, thereby simplifying the work that must be done by navigation algorithms. An example of a flight stabilizer system useful in various embodiments of UAVs according to the present invention is model Co-Pilot™ from FMA, Inc., of Frederick, Maryland. The Co-Pilot flight stabilizer system identifies a horizon with heat sensors, identifies changes in aircraft attitude relative to the horizon, and sends corrective signals to the servos (178) to keep the UAV flying straight and level.

15 UAV (100) includes an AVCS gyro (176). An AVCS gyro is an angular vector control system gyroscope that provides control signal to the servos to counter undesired changes in attitude such as those caused by sudden gusts of wind. An example of an AVCS gyro useful in various UAVs according to the present invention is model GYA350 from Futaba®.

20

Remote control devices according to embodiments of the present invention typically include automated computing machinery capable of receiving user selections of pixel on GUI maps, mapping the pixel to a waypoint location, receiving downlink telemetry including for example a starting position from a GPS receiver on the UAV, calculating a heading in dependence upon the starting position, the coordinates of the waypoint, and a navigation algorithm, identifying flight control instructions for flying the UAV on the heading, and transmitting the flight control instructions as uplink

25

telemetry from the remote control device to the UAV. Figure 3 is a block diagram of an exemplary remote control device showing relations among components of included automated computing machinery. In Figure 3, remote control device (161) includes a processor (164), also typically referred to as a central processing unit or  
5 'CPU.' The processor may be a microprocessor, a programmable control unit, or any other form of processor useful according to the form factor of a particular remote control device as will occur to those of skill in the art. Other components of remote control device (161) are coupled for data transfer to processor (164) through system bus (160).

10

Remote control device (161) includes random access memory or 'RAM' (166). Stored in RAM (166) an application program 152 that implements inventive methods of the present invention. In some embodiments, the application program (152) is OSGi compliant and therefore runs on an OSGi services framework installed (not  
15 shown) on a JVM (not shown). In addition, software programs and further information for use in implementing methods of navigating a UAV according to embodiments of the present invention may be stored in RAM or in non-volatile memory (168). Non-volatile memory (168) may be implemented as a magnetic disk drive such as a micro-drive, an optical disk drive, static read only memory ('ROM'),  
20 electrically erasable programmable read-only memory space ('EEPROM' or 'flash' memory), or otherwise as will occur to those of skill in the art.

Remote control device (161) includes communications adapter (170) implementing data communications connections (184) to other computers (162), including  
25 particularly computers on UAVs. Communications adapters implement the hardware level of data communications connections through which remote control devices communicate with UAVs directly or through networks. Examples of

communications adapters include modems for wired dial-up connections, Ethernet (IEEE 802.3) adapters for wired LAN connections, 802.11b adapters for wireless LAN connections, and Bluetooth adapters for wireless microLAN connections.

- 5 The example remote control device (161) of Figure 3 includes one or more input/output interface adapters (180). Input/output interface adapters in computers implement user-oriented input/output through, for example, software drivers and computer hardware for controlling output to display devices (184) such as computer display screens, as well as user input from user input devices (182) such as keypads,  
10 joysticks, keyboards, and touch screens.

Figure 4 sets forth a flow chart illustrating an exemplary method for navigating a UAV that includes receiving (402) in a remote control device a user's selection of a GUI map pixel (412) that represents a waypoint for UAV navigation. The pixel has a  
15 location on the GUI. Such a GUI map display has many pixels, each of which represents at least one position on the surface of the Earth. A user selection of a pixel is normal GUI operations to take a pixel location, row and column, from a GUI input/output adapter driven by a user input device such as a joystick or a mouse. The remote control device can be a traditional 'ground control station,' an airborne PDA  
20 or laptop, a workstation in Earth orbit, or any other control device capable of accepting user selections of pixels from a GUI map.

The method of Figure 4 includes mapping (404) the pixel's location on the GUI to Earth coordinates (414) of the waypoint. As discussed in more detail below with  
25 reference to Figure 5, mapping (404) the pixel's location on the GUI to Earth coordinates of the waypoint (414) typically includes mapping pixel boundaries of the GUI map to corresponding Earth coordinates and identifying a range of latitude and a

range of longitude represented by each pixel. Mapping (404) the pixel's location on the GUI to Earth coordinates of the waypoint (414) also typically includes locating a region on the surface of the Earth in dependence upon the boundaries, the ranges, and the location of the pixel on the GUI map.

5

The method of Figure 4 also includes receiving (408) downlink telemetry, including a starting position from a GPS receiver on the UAV, from the UAV through a socket on the remote control device. In fact, the remote control device is receiving downlink telemetry that includes a steady stream of GPS positions of the UAV. Receiving a  
10 starting position therefore is typically carried out by taking the current position of the UAV when the user selects the pixel as the starting position.

A socket is one end-point of a two-way communication link between two application programs running on a network. In Java, socket classes are used to represent a  
15 connection between a client program and a server program. The java.net package provides two Java classes--Socket and ServerSocket--that implement the client side of the connection and the server side of the connection, respectively. In some embodiments of the present invention, a Java web server, is included in an OSGi framework on a remote control device. Often then, a socket on the remote control  
20 device would be considered a server-side socket, and a socket on the UAV would be considered a client socket. In other embodiments of the present invention, a Java web server, is included in an OSGi framework on the UAV. In such embodiments, a socket on the UAV would be considered a server-side socket, and a socket on a remote control device would be considered a client socket.

25

Use of a socket requires creating a socket and creating data streams for writing to and reading from the socket. One way of creating a socket and two data streams for use with the socket is shown in the following exemplary pseudocode segment:

```
5      uavSocket = new Socket( "computerAddress", 7);  
      outStream = new PrintWriter(uavSocket.getOutputStream(), true);  
      inStream = new BufferedReader(new  
                               InputStreamReader(uavSocket.getInputStream()));
```

- 10 The first statement in this segment creates a new socket object and names it "uavSocket." The socket constructor used here requires a fully qualified IP address of the machine the socket is to connect to, in this case the Java server on a remote control device or a UAV, and the port number to connect to. In this example, "computerAddress" is taken as a domain name that resolves to a fully qualified dotted
- 15 decimal IP address. Alternatively, a dotted decimal IP address may be employed directly, as, for example, "195.123.001.001." The second argument in the call to the socket constructor is the port number. Port number 7 is the port on which the server listens in this example, whether the server is on a remote control device or on a UAV.
- 20 The second statement in this segment gets the socket's output stream and opens a Java PrintWriter object on it. Similarly, the third statement gets the socket's input stream and opens a Java BufferedReader object on it. To send data through the socket, an application writes to the PrintWriter, as, for example:

```
25      outStream.println(someWaypoint, macro, or Flight Control Instruction);
```



To receive data through the socket, an application reads from the `BufferedReader`, as show here for example:

a Waypoint, GPS data, macro, or flight control instruction = `inStream.readLine()`;

5

The method of Figure 4 also includes calculating (410) a heading in dependence upon the starting position, the coordinates of the waypoint, and a navigation algorithm. Methods of calculating a heading are discussed in detail below in this specification.

- 10 The method of Figure 4 includes identifying (418) flight control instructions for flying the UAV on the heading. Flight control instructions are specific commands that affect the flight control surfaces of the UAV. That is, instructions to move the flight control surfaces to affect the UAV's flight causing the UAV to turn, climb, descend, and so on. As an aid to further explanation, an exemplary method of
- 15 identifying flight control instructions for flying on a calculated heading is provided:

- receive new calculated heading from navigation algorithms
- read current heading from downlink telemetry
- if current heading is left of the calculated heading, identify flight control instruction: AILERONS LEFT 30 DEGREES
- if current heading is right of the calculated heading, identify flight control instruction: AILERONS RIGHT 30 DEGREES
- monitor current heading during turn

20

25

- when current heading matches calculated heading, identify flight control instruction: FLY STRAIGHT AND LEVEL

5 The method of Figure 4 includes transmitting (420) uplink telemetry, including the flight instructions, through the socket to the UAV. Transmitting (420) the flight control instructions from the remote control device to the UAV may be carried out by use of any data communications protocol, including, for example, transmitting the flight control instructions as form data, URI encoded data, in an HTTP message, a  
10 WAP message, an HDML message, or any other data communications protocol message as will occur to those of skill in the art.

Figure 4A is a data flow diagram illustrating an exemplary method for receiving downlink telemetry. The method of Figure 4A includes listening (450) on the socket  
15 (456) for downlink data (458). Listening on a socket for downlink data may be implemented by opening a socket, creating an input stream for the socket, and reading data from the input stream, as illustrated, for example, in the following segment of pseudocode:

```
20      uavSocket = new Socket( "computerAddress", 7);  
      inStream = new BufferedReader(new  
                               InputStreamReader(uavSocket.getInputStream()));  
      String downLinkData = inStream.readLine();
```

25 This segment opens a socket object named "uavSocket" with an input stream named "inStream." Listening for downlink data on the socket is accomplished with a

blocking call to `inStream.readLine()` which returns a String object name  
“downLinkData.”

5 The method of Figure 4A includes storing (452) downlink data (458) in computer  
memory (166) and exposing (454) the stored downlink data (458) through an API  
(462) to a navigation application (460). Downlink data typically is exposed through  
an ‘API’ (Application Programming Interface) by providing in a Java interface class  
public accessor functions for reading from member data elements in which the  
downlink data is stored. A navigation application wishing to access downlink data  
10 then may access the data by calling a public accessor methods, as, for example:  
`String someDownLinkData = APIimpl.getDownLinkData()`.

In the method of Figure 4A, the downlink telemetry (470) further comprises flight  
control instructions. It is counterintuitive that downlink telemetry contains flight  
15 control instruction when the expected data communications direction for flight  
control instructions ordinarily is in uplink from a remote control device to a UAV. It  
is useful to note, however, that flight control instructions can be uplinked from a  
multiplicity of remote control devices, not just one. A flight line technician with a  
handheld PDA can issue flight control instructions to a UAV that is also linked for  
20 flight control to a computer in a ground station. It is sometimes advantageous,  
therefore, for downlink telemetry to include flight control instructions so that one  
remote control device can be advised of the fact that some other remote control  
device issued flight control instructions to the same UAV.

25 Figure 4B sets forth a data flow diagram illustrating an exemplary method for  
transmitting uplink telemetry. The method of Figure 4B includes monitoring (466)  
computer memory (166) for uplink data (464) from a navigation application (464).

When uplink data (464) is presented, the method of Figure 4B includes sending (468) the uplink data through the socket (456) to the UAV (100). Sending uplink data through a socket may be implemented by opening a socket, creating an output stream for a socket, and writing the uplink data to the output stream, as illustrated, for example, in the following segment of pseudocode:

```
uavSocket = new Socket( "computerAddress", 7);  
outStream = new PrintWriter(uavSocket.getOutputStream(), true);  
outStream.println(String someUplinkData);
```

This segment opens a socket object named "uavSocket" with an output stream named "outStream." Sending uplink data through the socket is accomplished with a call to outStream.println() which takes as a call parameter a String object named "someUplinkData."

#### Macros

Although the flow chart of Figure 4 illustrates navigating a UAV to a single waypoint, as a practical matter, embodiments of the present invention typically support navigating a UAV along a route having many waypoints, including a final waypoint and one or more intermediate waypoints. That is, methods of the kind illustrated in Figure 4 may also include receiving user selections of a multiplicity of GUI map pixels representing waypoints, where each pixel has a location on the GUI and mapping each pixel location to Earth coordinates of a waypoint.

Such methods for navigating a UAV can also include assigning one or more UAV instructions to each waypoint and storing the coordinates of the waypoints and the

UAV instructions in computer memory on the remote control device. A UAV instruction typically includes one or more instructions for a UAV to perform a task in connection with a waypoint. Exemplary tasks include turning on or off a camera installed on the UAV, turning on or off a light installed on the UAV, orbiting a  
5      waypoint, or any other task that will occur to those of skill in the art.

UAV instructions to perform tasks in connection with a waypoint may be encoded in, for example, XML (the eXtensible Markup Language) as shown in the following exemplary XML segment:

10

```
<UAV-Instructions>
```

```
  <macro>
```

```
    <waypoint> 33° 44' 10'' N 30° 15' 50'' W </waypoint>
```

```
    <instruction> orbit </instruction>
```

15

```
    <instruction> videoCameraON </instruction>
```

```
    <instruction> wait30minutes </instruction>
```

```
    <instruction> videoCameraOFF </instruction>
```

```
    <instruction> nextWaypoint </instruction>
```

```
  </macro>
```

20

```
  <macro> </macro>
```

```
  <macro> </macro>
```

```
  <macro> </macro>
```

```
</UAV-instructions>
```

25      This XML example has a root element named 'UAV-instructions.' The example contains several subelements named 'macro.' One 'macro' subelement contains a waypoint location representing an instruction to fly to 33° 44' 10'' N 30° 15' 50'' W.

That macro subelement also contains several instructions for tasks to be performed when the UAV arrives at the waypoint coordinates, including orbiting around the waypoint coordinates, turning on an on-board video camera, continuing to orbit for thirty minutes with the camera on, turning off the video camera, and continuing to a  
5 next waypoint. Only one macro set of UAV instructions is shown in this example, but that is not a limitation of the invention. In fact, such sets of UAV instructions may be of any useful size as will occur to those of skill in the art.

Exemplary methods of navigating a UAV also include flying the UAV to each  
10 waypoint in accordance with one or more navigation algorithms and operating the UAV at each waypoint in accordance with the UAV instructions for each waypoint. Operating the UAV at the waypoint in accordance with the UAV instructions for each waypoint typically includes identifying flight control instructions in dependence upon the UAV instructions for each waypoint and transmitting the flight control  
15 instructions as uplink telemetry through a socket. Flight control instructions identified in dependence upon the UAV instructions for each waypoint typically include specific flight controls to move the flight control surfaces of the UAV causing the UAV to fly in accordance with the UAV instructions. For example, in the case of a simple orbit, a flight control instruction to move the ailerons and hold them at a  
20 certain position causing the UAV to bank at an angle can effect an orbit around a waypoint.

Operating the UAV at the waypoint in accordance with the UAV instructions for each way point typically includes transmitting the flight control instructions as uplink data  
25 from the remote control device to the UAV. Transmitting the flight control instructions as uplink data from the remote control device to the UAV may be carried out by use of any data communications protocol, including, for example, transmitting

the flight control instructions as form data, URI encoded data, in an HTTP message, a WAP message, an HDML message, or any other data communications protocol message as will occur to those of skill in the art.

5

### Pixel Mapping

For further explanation of the process of mapping pixels' locations to Earth coordinates, Figure 5 sets forth a block diagram that includes a GUI (502) displaying a map (not shown) and a corresponding area of the surface of the Earth (504). The GUI map has pixel boundaries identified as Row<sub>1</sub>, Col<sub>1</sub>; Row<sub>1</sub>, Col<sub>100</sub>; Row<sub>100</sub>, Col<sub>100</sub>; and Row<sub>100</sub>, Col<sub>1</sub>. In this example, the GUI map is assumed to include 100 rows of pixels and 100 columns of pixels. This example of 100 rows and columns is presented for convenience of explanation; it is not a limitation of the invention. GUI maps according to embodiments of the present invention may include any number of pixels as will occur to those of skill in the art.

The illustrated area of the surface of the Earth has corresponding boundary points identified as Lat<sub>1</sub>, Lon<sub>1</sub>; Lat<sub>1</sub>, Lon<sub>2</sub>; Lat<sub>2</sub>, Lon<sub>2</sub>; and Lat<sub>2</sub>, Lon<sub>1</sub>. This example assumes that the distance along one side of surface area (504) is 100 nautical miles, so that the distance expressed in terms of latitude or longitude between boundary points of surface area (504) is 100 minutes or 1° 40'.

In typical embodiments, mapping a pixel's location on the GUI to Earth coordinates of a waypoint includes mapping pixel boundaries of the GUI map to Earth coordinates. In this example, the GUI map boundary at Row<sub>1</sub>, Col<sub>1</sub> maps to the surface boundary point at Lat<sub>1</sub>, Lon<sub>1</sub>; the GUI map boundary at Row<sub>1</sub>, Col<sub>2</sub> maps to the surface boundary point at Lat<sub>1</sub>, Lon<sub>2</sub>; the GUI map boundary at Row<sub>2</sub>, Col<sub>2</sub> maps

to the surface boundary point at  $Lat_2, Lon_2$ ; the GUI map boundary at  $Row_2, Col_1$  maps to the surface boundary point at  $Lat_2, Lon_1$ .

- Mapping a pixel's location on the GUI to Earth coordinates of a waypoint typically also includes identifying a range of latitude and a range of longitude represented by each pixel. The range of latitude represented by each pixel may be described as  $(Lat_2 - Lat_1) / N_{rows}$ , where  $(Lat_2 - Lat_1)$  is the length in degrees of the vertical side of the corresponding surface (504), and  $N_{rows}$  is the number of rows of pixels. In this example,  $(Lat_2 - Lat_1)$  is  $1^\circ 40'$  or 100 nautical miles, and  $N_{rows}$  is 100 rows of pixels.
- The range of longitude represented by each pixel in this example therefore is one minute of arc or one nautical mile.

- Similarly, the range of longitude represented by each pixel may be described as  $(Lon_2 - Lon_1) / N_{cols}$ , where  $(Lon_2 - Lon_1)$  is the length in degrees of the horizontal side of the corresponding surface (504), and  $N_{cols}$  is the number of columns of pixels. In this example,  $(Lon_2 - Lon_1)$  is  $1^\circ 40'$  or 100 nautical miles, and  $N_{cols}$  is 100 columns of pixels. The range of longitude represented by each pixel in this example therefore is one minute of arc or one nautical mile.

- Mapping a pixel's location on the GUI to Earth coordinates of a waypoint typically also includes locating a region on the surface of the Earth in dependence upon the boundaries, the ranges, and the location of the pixel on the GUI map. The region is the portion of the surface corresponding the pixel itself. That region is located generally by multiplying in both dimension, latitude and longitude, the range of latitude and longitude by column or row numbers of the pixel location on the GUI map. That is, a latitude for the surface region of interest is given by Expression 1.



(Exp. 1) 
$$\text{Lat}_1 + P_{\text{row}} ((\text{Lat}_2 - \text{Lat}_1) / N_{\text{rows}})$$

In Expression 1:

5         $\text{Lat}_1$  is the latitude of an origin point for the surface area (504) corresponding generally to the GUI map,

$P_{\text{row}}$  is the row number of the pixel location on the GUI map, and

10         $((\text{Lat}_2 - \text{Lat}_1) / N_{\text{rows}})$  is the range of latitude represented by the pixel.

Similarly, a longitude for the surface region of interest is given by Expression 2.

(Exp. 2) 
$$\text{Lon}_1 + P_{\text{col}} ((\text{Lon}_2 - \text{Lon}_1) / N_{\text{cols}})$$

15

In Expression 2:

$\text{Lon}_1$  is the longitude of an origin point for the surface area (504) corresponding generally to the GUI map,

20

$P_{\text{col}}$  is the column number of the pixel location on the GUI map, and

$((\text{Lon}_2 - \text{Lon}_1) / N_{\text{cols}})$  is the range of longitude represented by the pixel.

25        Referring to Figure 5 for further explanation, Expressions 1 and 2 taken together identify a region (508) of surface area (504) that corresponds to the location of pixel (412) mapping the pixel location to the bottom left corner (506) of the region (508).

Advantageously, however, many embodiments of the present invention further map the pixel to the center of the region by adding one half of the length of the region's sides to the location of the bottom left corner (506).

- 5 More particularly, locating a region on the surface of the Earth in dependence upon the boundaries, the ranges, and the location of the pixel on the GUI map, as illustrated by Expression 3, may include multiplying the range of longitude represented by each pixel by a column number of the selected pixel, yielding a first multiplicand; and multiplying the range of longitude represented by each pixel by 0.5, yielding a second  
10 multiplicand; adding the first and second multiplicands to an origin longitude of the GUI map.

$$(Exp. 3) \quad Lon_1 + P_{col} ((Lon_2 - Lon_1) / N_{cols}) + 0.5((Lon_2 - Lon_1) / N_{cols})$$

- 15 In Expression 3, the range of longitude represented by each pixel is given by  $((Lon_2 - Lon_1) / N_{cols})$ , and the first multiplicand is  $P_{col} ((Lon_2 - Lon_1) / N_{cols})$ . The second multiplicand is given by  $0.5((Lon_2 - Lon_1) / N_{cols})$ .

- Similarly, locating a region on the surface of the Earth in dependence upon the  
20 boundaries, the ranges, and the location of the pixel on the GUI map, as illustrated by Expression 4, typically also includes multiplying the range of latitude represented by each pixel by a row number of the selected pixel, yielding a third multiplicand; multiplying the range of latitude represented by each pixel by 0.5, yielding a fourth multiplicand; and adding the third and fourth multiplicands to an origin latitude of the  
25 GUI map.

$$(Exp. 4) \quad Lat_1 + P_{row} ((Lat_2 - Lat_1) / N_{rows}) + 0.5((Lat_2 - Lat_1) / N_{rows})$$

In Expression 4, the range of latitude represented by each pixel is given by  $((\text{Lat}_2 - \text{Lat}_1) / N_{\text{rows}})$ , and the third multiplicand is  $P_{\text{row}} ((\text{Lat}_2 - \text{Lat}_1) / N_{\text{rows}})$ . The fourth multiplicand is given by  $0.5((\text{Lat}_2 - \text{Lat}_1) / N_{\text{rows}})$ . Expressions 3 and 4 taken together  
 5 map the location of pixel (412) to the center (510) of the located region (508).

### Navigation on a Heading to a Waypoint

An exemplary method of navigating in accordance with a navigation algorithm is  
 10 explained with reference to Figures 6 and 7. Figure 6 sets forth a flow chart illustrating an exemplary method of navigating a UAV in accordance with a navigation algorithm, and Figure 7 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 6.

15 The method of Figure 6 includes periodically repeating (610) the steps of, receiving (602) in the remote control device from the GPS receiver a current position of the UAV, and calculating (604) a new heading from the current position to the waypoint. The method of Figure 6 also includes identifying (606) flight control instructions for flying the UAV on the new heading, and transmitting (608), from the remote control  
 20 device to the UAV, the flight control instructions for flying the UAV on the new heading. In this method, if  $\text{Lon}_1$ ,  $\text{Lat}_1$  is taken as the current position, and  $\text{Lon}_2$ ,  $\text{Lat}_2$  is taken as the waypoint position, then the new heading may be calculated generally as the inverse tangent of  $((\text{Lat}_2 - \text{Lat}_1) / (\text{Lon}_2 - \text{Lon}_1))$ .

25 Figure 7 shows the effect of the application of the method of Figure 6. In the example of Figure 7, a UAV is flying in a cross wind having cross wind vector (708). Curved flight path (716) results from periodic calculations according to the method of

Figure 6 of a new heading straight from a current location to the waypoint. Figure 7 shows periodic repetitions of the method of Figure 6 at plot points (710, 712, 714). For clarity of explanation, only three periodic repetitions are shown, although that is not a limitation of the invention. In fact, any number of periodic repetitions may be used as will occur to those of skill in the art.

#### Navigation with Headings set to a Cross Track Direction

A further exemplary method of navigating in accordance with a navigation algorithm is explained with reference to Figures 8 and 9. Figure 8 sets forth a flow chart illustrating an exemplary method of navigating a UAV in accordance with a navigation algorithm, and Figure 9 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 8. The method of Figure 8 includes identifying (802) a cross track between the starting point and the waypoint. A cross track is a fixed course from a starting point directly to a waypoint. If  $Lon_1, Lat_1$  is taken as the position of a starting point, and  $Lon_2, Lat_2$  is taken as the waypoint position, then a cross track is identified by  $Lon_1, Lat_1$  and  $Lon_2, Lat_2$ . A cross track has a direction, a 'cross track direction,' that is the direction straight from a starting point to a waypoint, and it is often useful to characterize a cross track by its cross track direction. The cross track direction for a cross track identified by starting point  $Lon_1, Lat_1$  and waypoint position  $Lon_2, Lat_2$  may be calculated generally as the inverse tangent of  $((Lat_2 - Lat_1) / (Lon_2 - Lon_1))$ .

The method of Figure 8 includes periodically repeating (810) the steps of receiving (804) in the remote control device from the GPS receiver a current position of the UAV, and calculating (806) a shortest distance between the current position and the cross track. If the shortest distance between the current position and the cross track is

greater than a threshold distance (808), the method of Figure 8 includes transmitting (812) flight control instructions that pilot the UAV toward the cross track, and, when the UAV arrives at the cross track, transmitting (814) flight control instructions that pilot the UAV in a cross track direction toward the waypoint.

5

Figure 9 illustrates calculating a shortest distance between the current position and a cross track. In the example of Figure 9, calculating a shortest distance between the current position and a cross track includes calculating the distance from a current position (912) to the waypoint (704). In the example of Figure 9, the distance from the current position (912) to the waypoint (704) is represented as the length of line (914). For current position  $Lon_1$ ,  $Lat_1$  and waypoint position  $Lon_2$ ,  $Lat_2$ , the distance from a current position (912) to the waypoint (704) is given by the square root of  $(Lat_2 - Lat_1)^2 + (Lon_2 - Lon_1)^2$ .

15 In this example, calculating a shortest distance between the current position and a cross track also includes calculating the angle (910) between a direction from the current position (912) to the waypoint (704) and a cross track direction. In the example of Figure 9, the direction from the current position (912) to the waypoint (704) is represented as the direction of line (914). In the example of Figure 9, the cross track direction is the direction of cross track (706). The angle between a direction from the current position to the waypoint and a cross track direction is the difference between those directions.

25 In the current example, calculating a shortest distance between the current position and a cross track also includes calculating the tangent of the angle between a direction from the current position to the waypoint and a cross track direction and multiplying the tangent of the angle by the distance from the current position to the waypoint.

Figure 9 also shows the effect of the application of the method of Figure 8. In the example of Figure 9, a UAV is flying in a cross wind having cross wind vector (708). The flight path (904) results from periodic calculations according to the method of Figure 8 of a shortest distance between a current position and the cross track (706), flying the UAV back to the cross track and then flying in the direction of the cross track whenever the distance from the cross track exceeds a predetermined threshold distance.

10                    Headings set to Cross Track Direction with Angular Thresholds

A further exemplary method of navigating in accordance with a navigation algorithm is explained with reference to Figures 10 and 11. Figure 10 sets forth a flow chart illustrating an exemplary method of navigating a UAV in accordance with a navigation algorithm, and Figure 11 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 10.

In the method of Figure 10, piloting in accordance with a navigation algorithm includes identifying (1002) a cross track having a cross track direction between the starting point and the waypoint. As described above, a cross track is identified by a position of a starting point and a waypoint position. For a starting point position of  $Lon_1, Lat_1$  and a waypoint position of  $Lon_2, Lat_2$ , a cross track is identified by  $Lon_1, Lat_1$  and  $Lon_2, Lat_2$ . In addition, it is often also useful to characterize a cross track by its cross track direction. The cross track direction for a cross track identified by starting point  $Lon_1, Lat_1$  and waypoint position  $Lon_2, Lat_2$  may be calculated generally as the inverse tangent of  $((Lat_2 - Lat_1) / (Lon_2 - Lon_1))$ .

In the method of Figure 10, navigating a UAV in accordance with a navigation algorithm includes periodically repeating (1010) the steps of receiving (1004) in the remote control device from the GPS receiver a current position and a current heading of the UAV, and calculating (1006) an angle between the direction from the current position to the waypoint and a cross track direction. If the angle is greater than a threshold angle (1008), the method of Figure 10 includes transmitting (1012) flight control instructions that pilot the UAV toward the cross track, and, upon arriving at the cross track, transmitting (1014) flight control instructions that pilot the UAV in the cross track direction toward the waypoint.

10

Transmitting (1012) flight control instructions that pilot the UAV toward the cross track is carried out by transmitting flight control instructions to turn to a heading no more than ninety degrees from the cross track direction, turning to the left if the current position is right of the cross track and to the right if the current position is left of the cross track. Transmitting (1014) flight control instructions that pilot the UAV in the cross track direction toward the waypoint transmitting flight control instructions to turn the UAV to the cross track direction and then flying straight and level on the cross track direction.

15

Figure 11 shows the effect of the application of the method of Figure 10. In the example of Figure 11, a UAV is flying in a cross wind having cross wind vector (708). The flight path (1104) results from periodically transmitting flight control instructions to fly the UAV, according to the method of Figure 10, back to the cross track and then in the direction of the cross track whenever an angle between the direction from the current position to the waypoint and a cross track direction exceeds a predetermined threshold angle.

20  
25

In many embodiments of the method of Figure 10, the threshold angle is a variable whose value varies in dependence upon a distance between the UAV and the waypoint. In typical embodiments that vary the threshold angle, the threshold angle is increased as the UAV flies closer to the waypoint. It is useful to increase the  
5 threshold angle as the UAV flies closer to the waypoint to reduce the risk of excessive 'hunting.' That is, because the heading is the cross track direction, straight to the WP rather than cross-wind, if the angle remains the same, the distance that the UAV needs to be blown off course to trigger transmitting flight control signals instructing the UAV to return to the cross track gets smaller and smaller until the  
10 UAV is flying to the cross track, turning to the cross track direction, getting blown immediately across the threshold, flying back the cross track, turning to the cross track direction, getting blown immediately across the threshold, and so on, and so on, in rapid repetition. Increasing the threshold angle as the UAV flies closer to the waypoint increases the lateral distance available for wind error before triggering the  
15 transmission of flight instructions to return to the cross track, thereby reducing this risk of excessive hunting.

Figure 12 sets forth a flow chart illustrating an exemplary method for navigating a UAV that includes receiving (402) in a remote control device a user's selection of a  
20 GUI map pixel (412) that represents a waypoint for UAV navigation. The pixel has a location on the GUI. Such a GUI map display has many pixels, each of which represents at least one position on the surface of the Earth. A user selection of a pixel is normal GUI operations to take a pixel location, row and column, from a GUI input/output adapter driven by a user input device such as a joystick or a mouse. The  
25 remote control device can be a traditional 'ground control station,' an airborne PDA or laptop, a workstation in Earth orbit, or any other control device capable of accepting user selections of pixels from a GUI map.



The method of Figure 12 includes mapping (404) the pixel's location on the GUI to Earth coordinates of the waypoint (414). As discussed in more detail below with reference to Figure 5, mapping (404) the pixel's location on the GUI to Earth  
5 coordinates of the waypoint (414) typically includes mapping pixel boundaries of the GUI map to corresponding Earth coordinates and identifying a range of latitude and a range of longitude represented by each pixel. Mapping (404) the pixel's location on the GUI to Earth coordinates of the waypoint (414) also typically includes locating a region on the surface of the Earth in dependence upon the boundaries, the ranges, and  
10 the location of the pixel on the GUI map.

The method of Figure 12 also includes transmitting (406) uplink telemetry, including the coordinates of the waypoint, to the UAV through a socket on the remote control device. Transmitting (406) uplink telemetry, including the coordinates of the  
15 waypoint, to the UAV through a socket on the remote control device may be carried out by use of any data communications protocol, including, for example, transmitting the coordinates as form data, URI encoded data, in an HTTP message, a WAP message, an HDML message, or any other data communications protocol message as will occur to those of skill in the art. Transmitting uplink telemetry through a socket  
20 may be implemented by opening a socket, creating an output stream for the socket, and writing uplink telemetry data to the output stream, as illustrated, for example, in the following segment of pseudocode:

```
25      uavSocket = new Socket( "computerAddress", 7);  
      outputStream = new PrintWriter(uavSocket.getOutputStream(), true);  
      outputStream.println(String someUplinkData);
```

This segment opens a socket object named “uavSocket” with an output stream named “outStream.” Transmitting uplink telemetry through the socket is accomplished with a call to `outStream.println()` which takes as a call parameter a String object named “someUplinkData.”

5

The method of Figure 12 also includes receiving (408) downlink telemetry, include a starting position from a GPS receiver, from the UAV through the socket and piloting (410) the UAV, under control of a navigation computer on the UAV, from the starting position to the waypoint in accordance with a navigation algorithm. Methods of piloting a UAV according to a navigation algorithm are discussed in detail below in this specification.

Receiving downlink telemetry through a socket may be implemented by opening a socket, creating an input stream for the socket, and reading data from the input stream, as illustrated, for example, in the following segment of pseudocode:

15

```
uavSocket = new Socket( "computerAddress", 7);
inStream = new BufferedReader(new
                                InputStreamReader(uavSocket.getInputStream()));
String downLinkTelemetry = inStream.readLine();
```

20

This segment opens a socket object named “uavSocket” with an input stream named “inStream.” Receiving downlink telemetry through the socket is accomplished with a blocking call to `inStream.readLine()` which returns a String object name “downLinkTelemetry.”

25

In the method of Figure 12, downlink telemetry may include Earth coordinates of

waypoints as well as one or more UAV instructions. It is counterintuitive that downlink telemetry contains waypoint coordinates and UAV instructions when the expected data communications direction for waypoint coordinates and UAV instructions ordinarily is in uplink from a remote control device to a UAV. It is  
5 useful to note, however, that waypoint coordinates and UAV instructions can be uplinked from a multiplicity of remote control devices, not just one. A flight line technician with a handheld PDA can issue waypoint coordinates and UAV instructions to a UAV that is also linked for flight control to a computer in a ground station. It is sometimes advantageous, therefore, for downlink telemetry to include  
10 waypoint coordinates or UAV instructions so that one remote control device can be advised of the fact that some other remote control device issued waypoint coordinates or UAV instructions to the same UAV.

#### Macros

15 As mentioned above, embodiments of the present invention often support navigating a UAV along a route having many waypoints, including a final waypoint and one or more intermediate waypoints. That is, methods of the kind illustrated in Figure 12 may also include receiving user selections of a multiplicity of GUI map pixels  
20 representing waypoints, where each pixel has a location on the GUI and mapping each pixel location to Earth coordinates of a waypoint.

Such methods of navigating a UAV can also include assigning one or more UAV instructions to each waypoint and transmitting the coordinates of the waypoints and  
25 the UAV instructions in the uplink telemetry through the socket to the UAV. A UAV instruction typically includes one or more instructions for a UAV to perform a task in connection with a waypoint. Exemplary tasks include turning on or off a camera

installed on the UAV, turning on or off a light installed on the UAV, orbiting a waypoint, or any other task that will occur to those of skill in the art. Such exemplary methods of navigating a UAV also include storing the coordinates of the waypoints and the UAV instructions in computer memory on the UAV, piloting the UAV to  
5 each waypoint in accordance with one or more navigation algorithms, and operating the UAV at each waypoint in accordance with the UAV instructions for each waypoint.

#### Navigation on a Course to a Waypoint

10

A further exemplary method of navigating in accordance with a navigation algorithm is explained with reference to Figures 13, 14, and 15. Figure 13 sets forth a flow chart illustrating an exemplary method of piloting in accordance with a navigation algorithm. Figure 14 sets forth a line drawing illustrating a method of calculating a  
15 heading with a cross wind to achieve a particular ground course. And Figure 15 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 13.

In the method of Figure 13, piloting in accordance with a navigation algorithm  
20 comprises periodically repeating (1212) the steps of reading (1202) from the GPS receiver a current position of the UAV; calculating (1204) a direction to the waypoint from the current position; calculating (1206) a heading in dependence upon wind speed, wind direction, air speed, and the direction to the waypoint; turning (1208) the UAV to the heading; and flying (1210) the UAV on the heading.

25

Figure 14 illustrates calculating (1206) a heading in dependence upon wind speed, wind direction, air speed, and the direction to the waypoint. Figure 14 sets forth a

line drawing illustrating relations among several pertinent vectors, a wind velocity (1222), a resultant velocity (1224) , and a UAV's air velocity (1226). A velocity vector includes a speed and a direction. These vectors taken together represent wind speed, wind direction, air speed, and the direction to the waypoint. In the example of Figure 14, the angle B is a so-called wind correction angle, an angle which subtracted from (or added to, depending on wind direction) a direction to a waypoint yields a heading, a compass heading for a UAV to fly so that its resultant ground course is on a cross track. A UAV traveling at an air speed of 'a' on heading (D – B) in the presence of a wind speed 'b' with wind direction E will have resultant ground speed 'c' in direction D.

In Figure 14, angle A represents the difference between the wind direction E and the direction to the waypoint D. In Figure 14, the wind velocity vector (1222) is presented twice, once to show the wind direction as angle E and again to illustrate angle A as the difference between angles E and D. Drawing wind velocity (1222) to form angle A with the resultant velocity (1224) also helps explain how to calculate wind correction angle B using the law of sines. Knowing two sides of a triangle and the angle opposite one of them, the angle opposite the other may be calculated, in this example, by  $B = \sin^{-1}(b (\sin A) / a)$ . The two known sides are airspeed 'a' and wind speed 'b.' The known angle is A, the angle opposite side 'a,' representing the difference between wind direction E and direction to the waypoint D. Calculating a heading, angle F on Figure 14, is then carried out by subtracting the wind correction angle B from the direction to the waypoint D.

Figure 15 shows the effect of the application of the method of Figure 13. In the example of Figure 15, a UAV is flying in a cross wind having cross wind vector (708). Curved flight path (1316) results from periodic calculations according to the

method of Figure 13 of a new heading straight whose resultant with a wind vector is a course straight from a current location to the waypoint. Figure 15 shows periodic repetitions of the method of Figure 13 at plot points (1310, 1312, 1314). For clarity of explanation, only three periodic repetitions are shown, although that is not a  
5 limitation of the invention. In fact, any number of periodic repetitions may be used as will occur to those of skill in the art.

Navigation on a Course set to a Cross Track Direction

10 A further exemplary method of navigating in accordance with a navigation algorithm is explained with reference to Figures 16 and 17. Figure 16 sets forth a flow chart illustrating an exemplary method of piloting in accordance with a navigation algorithm, and Figure 17 sets forth a line drawing illustrating a flight path produced by application of the method of Figure 16.

15

The method of Figure 16 includes identifying (1402) a cross track and calculating (1404) a cross track direction from the starting position to the waypoint. In the method of Figure 16, piloting in accordance with a navigation algorithm is carried out by periodically repeating the steps of reading (1406) from the GPS receiver a current  
20 position of the UAV; calculating (1408) a shortest distance between the cross track and the current position; and, if the shortest distance between the cross track and the current position is greater than a threshold distance, piloting (1412) the UAV to the cross track. Upon arriving at the cross track, the method includes: reading (1414) from the GPS receiver a new current position of the UAV; calculating (1416), in  
25 dependence upon wind speed, wind direction, air speed, and the cross track direction, a new heading; turning (1418) the UAV to the new heading; and flying (1420) the UAV on the new heading.

Figure 17 shows the effect of the application of the method of Figure 16. In the example of Figure 17, a UAV is flying in a cross wind having cross wind vector (708). Curved flight path (1304) results from periodic calculations according to the method of Figure 16 of a shortest distance between a current position and the cross track (706), flying the UAV back to the cross track, and, upon arriving at the cross track, calculating a new heading and flying the UAV on the new heading.

It will be understood from the foregoing description that modifications and changes may be made in various embodiments of the present invention without departing from its true spirit. The descriptions in this specification are for purposes of illustration only and are not to be construed in a limiting sense. The scope of the present invention is limited only by the language of the following claims.